TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol.12, No.5, May 2014, pp. 3281 ~ 3288
DOI: http://dx.doi.org/10.11591/telkomnika.v12i5.4947

A New License Plate Fault-tolerant Characters Recognition Algorithm

Guowei Yang*[1], Xiaochun Wang[1], Yang Yang[2]
[1]College of Automation Engineering, Qingdao University,
No.308, Ningxia Road, Qingdao, Shandong, 266071, China, Ph./Fax: 0532-85956069
[2]Qingzhou Cigarette Factory, Shandong Tobacco Industry Co., Ltd,
No.1818, Linglongshanbei Road, Qingzhou, Shandong, 262500, China, Ph./Fax: 0531-82599001
Corresponding author, e-mail: ygw_ustb@163.com, wangxiaochun_yt@126.com, yy_bonnie@163.com

Abstract

According to the license plate recognition problem, this paper did the research about license plate location and characters recognition. It proposed two new algorithms, they separately are license location algorithm based on color segmentation and fault-tolerant characters recognition algorithm based on BP neural network. In the pre-processing stage, it proposed image enhancement algorithm which could make the image more easily analyzed by computer. In the location stage, it made utilization of color and shape information, and then proposed location algorithm. In the recognition stage, it fully made the consideration of characters' fault-tolerant, and then made the use of improved BP neural network to recognize characters. It did some experiment by MATLAB. Experiments show that the special license plate fault-tolerant characters recognition algorithm is more accurate than the original license plate recognition methods, and its recognition rate has been improved greatly.

Keywords: *characters recognition, color segmentation, fault-tolerant, BP neural network, MATLAB*

1. Introduction

Intelligent transportation system has become an important direction of current traffic management and development, and license plate recognition as a kind of traffic information acquisition technology has increasingly attracted more and more attention. It can automatically authenticate the identity of the vehicle, and make vehicle management, traffic flow control and intersection payment highly automated, so it has a wide range of practical application.

For a license plate recognition system, the recognition process generally includes the following steps: license plate image preprocessing, location, character segmentation, feature extraction, character recognition and post-processing. Firstly, using the color segmentation technology find the possible region through color histogram, then test the length-width ratio, length, height and plate texture of the region to obtain the best location. Secondly, did the character segmentation and feature vectors extraction. Thirdly, put the vectors to the improved BP neural network for training. Lastly, recognized the characters. Figure 1 is license plate recognition system.

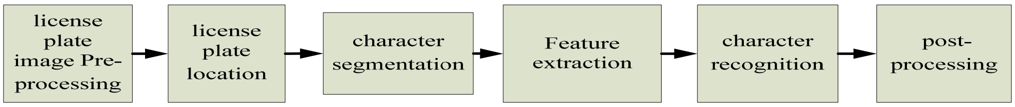

Figure 1. License Plate Recognition System Block Diagram

2. License Plate Image Pre-processing

2.1. Image Enhancement

Currently, there are several license plate location algorithms [1, 2]. When using the histogram equalization method to adjust the brightness of the license plate image, we find that

Received October 10, 2013; Revised November 27, 2013; Accepted December 14, 2013

light and dark contrast between the characters and background in the image is weakened, which will reduce the accuracy of location and recognition. So we adapt the following steps to adjust the brightness of the plate image gradually.

Statistic the number of blue pixels of the image, then fix the value of a and b to meet the requirements that the number of pixels which brightness value x∈[min,a] accounts for 5% of the total number of blue pixels and the number of pixels which brightness value x∈[b,max] also accounts for 5% of the total number of blue pixels.

Set the brightness value which is less than min to a, and the brightness value which is greater than max to b.

Similarly, do the same steps to the red and green pixels. From doing above , we accomplish the brightness adjustment.

After the brightness adjustment, we should do the filtering to make the further enhancement.

2.2. License Plate Location

In the process of image analysis and processing, it is considerable to select an appropriate threshold to separate the target from background. In this paper, a new threshold segmentation algorithm is as follows:

Assume consider R component firstly. L means the gray scale of the image, n_i means the number of pixels whose gray value are i, and N means the total number of pixels.

$$N = \sum_{i=1}^{L} \mathrm{n}_i \tag{1}$$

Calculate pi:

$$p_i = {n_i}/{N}, p_i \geq 0, \sum_{i=1}^{L} p_i = 1 \tag{2}$$

The image is divided into two parts. They are C0 (target)and C1(background). C0 means pixels whose gray scale are [1,…,k], and C1 means pixels whose gray scale are [k+1,…,L].

$$w_0 = \sum_{i=1}^{k} p_i = w(k) \tag{3}$$

$$w_1 = \sum_{i=k+1}^{L} p_i = 1 - w(k) \tag{4}$$

Average gray scale of C0 and C1 are:

$$\mathrm{u}_0 = \sum_{i=1}^{i=k} ip_i \Big/ w_0 \tag{5}$$

$$\mathrm{u}_1 = \sum_{i=k+1}^{i=L} ip_i \Big/ w_1 \tag{6}$$

Average gray scale of the imagine is:

$$\mathrm{u}_T = \sum_{i=1}^{i=L} ip_i = w_0 \cdot u_0 + w_1 \cdot u_1 \tag{7}$$

Within-class variance of C0 and C1 are:

$$\sigma_0^2 = \sum_{i=1}^{i=k} (1-u_0)^2 p_i \Big/ w_0 \tag{8}$$

$$\sigma_1^2 = \sum_{i=k+1}^{i=L} (1-u_1)^2 p_i \Big/ w_1 \tag{9}$$

Between-class variance of C0 and C1 and population variance are:

$$\begin{aligned} \delta_B^2 &= w_0(u_0-u_T)^2 + w_1(u_1-u_T)^2 \\ &= w_0 w_1 (u_1-u_0)^2 \end{aligned} \tag{10}$$

$$\delta_T^2 = \sum_{i=1}^{i=L} (i-u_T)^2 p_i \tag{11}$$

Set parameter η:

$$\eta = \delta_B^2 \big/ \delta_T^2 \tag{12}$$

When η takes to the maximum, the corresponding k value is the best threshold.Similarly, do the same calculation to G and B component, then get the best threshold. In China, there are three kinds of ratio: police cars and military vehicle (white background) is 3.8, large vehicle (back plate) is 2.0, and the remaining vehicles is 3.6. We select the ratio around 3.6 because we mainly study the blue-white plate. Figure 2 are the location images based on color segmentation.

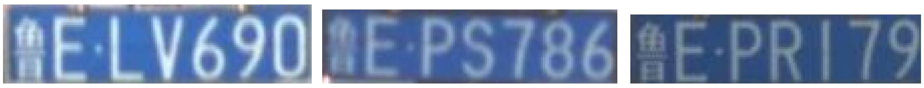

Figure 2. The License Plate Location Based on Color Segmentation

3. Characters Recognition Based on Feature Extraction and BP Neural Network

3.1. Character Segmentation

Firstly, transform the location image into binary image which background is back and target is white. Plate region binarization is good or bad which directly impact the accuracy of character segmentation and recognition. There are several traditional character segmentation methods such as horizontal projection [3], template matching [4], cluster analysis [5], etc. We adapt the horizontal projection method horizontal projection.

3.2. Character Image Normalization

The characters segmented from the plate image are not in the same size. In order to recognize the characters conveniently, we should turn them into the same size. It adapt bilinear interpolation algorithm to transform these character images into 20*35. Assume H as the original image's horizontal projection and V mean the original image's vertical projection. M means the heighth after normalization and N means the width after normalization. Point(m,n) after normalization is expressed as [6]:

$$\mathrm{m} = \sum_{k=1}^{i} H(k) \times \frac{M}{\sum_{k=1}^{j} H(k)} \tag{13}$$

$$\mathrm{n} = \sum_{k=1}^{i} V(k) \times \frac{N}{\sum_{k=1}^{j} V(k)} \tag{14}$$

3.3. Character Feature Extraction

It adapt coarse grid feature extraction in this paper. The basic idea of coarse grid feature extraction is: equally divide the character image which is after size and location normalization into M*N grids, then statistics the number of white pixels in every grids. If the number of white pixels over 20% of total pixels, set feature value of this grid to1, otherwise to 0. In this paper, we normalize these characters into 70*50, then divide them into 7*5 grids. So we can get the feature vector of each character, they are all 35-dimensional vectors which are composed by 0 and 1.

3.4. Character Recognition

3.4.1. Character Fault-tolerant

We adapt the method which combined feature extraction and BP neural network. When recognizing the characters, take fully account of character fault-tolerant. And calculate noisy samples which are theoretically allowed of each character. Take the Chinese characters for example to explain. Table 1 is feature vectors of the several Chinese characters from the 31 Chinese characters.

Table 1. Feature Vectors of the Several Chinese Characters

藏	1	1	1	1	1	0	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	0	0	0	0	0
川	0	0	0	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	0	0	1	0
鄂	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	0	1	1	1	0
赣	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
贵	0	0	0	0	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	0	1	1
桂	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0	0	1	0	1	1
黑	0	1	1	1	0	0	1	1	1	1	0	1	1	1	1	0	1	1	1	0	0	1	1	1	0	1	1	1	1	1	0	0	0	0	0
沪	1	1	1	1	0	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0	0	1	1	1	0	0	1	1	1	0	0
吉	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	0	0	1	1	1	1	1	1	1
冀	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	0	0	0	0	1
津	1	0	1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	0
晋	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	0	0	0	1	0	1	0	1	1	1	1	0	0	1	1	1	0
京	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	1	1	0	1
辽	1	1	1	1	1	0	1	0	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1
鲁	0	1	0	0	0	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1
蒙	0	1	1	1	0	0	1	1	1	0	0	0	1	0	0	0	1	1	0	0	0	1	1	1	0	0	1	1	1	1	0	0	1	0	0
闽	0	0	1	1	0	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	1	0	0	0	0	0
琼	0	0	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1
陕	0	0	0	0	0	1	1	1	1	0	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1
苏	1	1	1	1	1	0	1	1	1	0	1	1	1	1	0	0	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	1	0
湘	0	0	1	1	0	0	0	1	1	1	0	0	1	1	1	0	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	0	0	0	0
新	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1
渝	0	0	0	0	0	1	0	1	1	0	0	1	1	1	0	0	1	1	0	0	0	1	1	1	0	1	1	1	1	1	1	1	1	0	0
豫	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	1	0
粤	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	0	0	1	1	0
云	0	1	1	1	0	0	1	1	1	0	0	0	0	0	0	1	1	1	1	1	0	1	1	0	0	1	1	0	1	0	1	1	1	1	1
浙	1	0	1	1	1	0	1	1	1	0	0	0	1	1	1	0	1	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	0	1
青	0	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1
皖	1	1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
宁	1	1	1	1	0	1	1	0	1	1	1	1	1	1	1	0	0	1	1	0	0	0	1	1	0	0	0	1	1	0	0	1	1	0	0
甘	0	1	0	1	1	0	1	1	1	1	0	1	0	1	1	0	1	1	1	1	0	1	0	1	1	0	1	0	1	1	0	1	1	1	0

The number of different bits of feature vector between”藏’and others are:
7,10,11,10,7,8,11,12,12,4,10,7,6,13,11,12,14,12,8,13,8,15,8,8,15,14,9,11,13,17;

The number of different bits of feature vector between “川’and others are :

7,13,12,3,8,11,12,13,7,11,13,7,9,10,12,13,9,7,11,14,22,20,11,11,14,17,11,12,16,18；

The number of different bits of feature vector between “鄂’and others are :

10,13,3,14,7,12,9,6,10,8,8,9,10,7,15,12,8,14,6,13,4,17,2,2,17,12,6,5,11,9;

The number of different bits of feature vector between “赣’and others are :

11,12,3,11,6,13,8,3,9,7,9,6,7,6,18,13,5,11,7,14,3,16,3,5,14,9,3,2,14,12;

The number of different bits of feature vector between “贵’and others are :

10,3,14,11,6,13,13,12,8,12,14,7,8,7,17,16,8,6,12,17,10,9,14,14,13,16,10,11,17,19;

The number of different bits of feature vector between “桂’and others are :

7,8,7,6,7,11,10,9,5,11,10,6,7,6,16,12,9,9,9,14,3,16,7,7,14,13,7,8,12,16;

The number of different bits of feature vector between “黑’and others are :

8,11,12,13,14,11,13,16,8,10,12,11,14,13,7,10,12,14,14,8,12,11,10,12,15,14,10,15,11,10;

The number of different bits of feature vector between “沪’and others are :

11,12,9,8,13,10,13,11,11,9,11,8,11,12,16,15,11,13,11,16,11,16,9,9,12,15,11,10,11,18;

The number of different bits of feature vector between “吉’and others are :

12,13,6,3,12,9,16,11,10,10,12,7,8,9,19,14,8,13,8,15,6,17,6,8,13,10,6,3,15,13;

The number of different bits of feature vector between “冀’and others are :

12,7,10,9,8,5,8,11,10,10,12,3,8,9,11,12,10,10,12,13,6,13,8,10,15,14,10,9,13,19;

The number of different bits of feature vector between “津’and others are :

4,11,8,7,12,11,10,9,10,10,14,9,12,9,15,8,4,8,10,9,10,13,6,8,17,10,8,9,15,15;

The number of different bits of feature vector between “晋’and others are :

10,13,8,9,14,10,12,11,12,12,14,11,8,14,15,18,14,16,12,19,10,15,8,8,17,18,12,11,9,13;

The number of different bits of feature vector between “京’and others are :

7,7,9,6,7,6,11,8,7,3,9,11,5,8,12,15,9,9,9,16,7,10,9,11,12,11,9,6,12,18;

The number of different bits of feature vector between “辽’and others are :

6,9,10,7,8,7,14,11,8,8,12,8,5,11,15,18,12,13,6,19,8,14,10,10,11,12,8,7,13,17;

The number of different bits of feature vector between “鲁’and others are :

13,10,7,6,7,6,13,12,9,9,9,14,8,11,16,13,5,7,11,14,5,12,7,9,14,13,7,8,17,12;

The number of different bits of feature vector between “蒙’and others are :

11,12,15,18,17,16,7,16,19,11,15,15,12,15,16,17,18,17,13,12,15,10,15,15,12,15,17,18,14,14;

The number of different bits of feature vector between “闽’and others are :

12,13,12,13,16,12,10,15,14,12,8,18,15,18,13,17,10,14,16,7,12,17,10,12,19,14,10,13,17,15;

The number of different bits of feature vector between “琼’and others are :

14,9,8,5,8,9,12,11,8,10,4,14,9,12,5,18,10,6,12,11,8,13,6,8,15,12,6,7,17,15;

The number of different bits of feature vector between “陕’and others are :

12,7,14,11,6,9,14,13,13,10,8,16,9,13,7,17,14,6,14,15,12,11,12,12,17,14,12,13,17,19;

The number of different bits of feature vector between “苏’and others are :

8,11,6,7,12,9,14,11,8,12,10,12,9,6,11,13,16,12,14,15,8,15,8,5,13,9,8,7,13,11;

The number of different bits of feature vector between “湘’and others are :

13,14,13,14,17,14,8,16,15,13,9,19,16,19,14,12,7,11,15,15,13,14,11,13,18,9,11,16,16,14;

The number of different bits of feature vector between “新’and others are :

8,11,4,3,10,3,12,11,6,6,10,10,7,8,5,15,12,8,12,8,13,17,4,6,15,10,4,5,15,13;

The number of different bits of feature vector between “渝’and others are :

15,10,17,16,9,16,11,16,17,13,13,15,10,14,12,10,17,13,11,15,14,17,17,19,16,15,17,16,16,16;

The number of different bits of feature vector between “豫’and others are :

8,11,2,3,14,7,10,9,6,8,6,8,9,10,7,15,10,6,12,8,11,4,17,2,17,12,6,5,13,11;

The number of different bits of feature vector between “粤’and others are :

8,11,2,5,14,7,12,9,8,10,8,8,11,10,9,15,12,8,12,5,13,6,19,2,17,14,8,7,11,11,;

The number of different bits of feature vector between “云’and others are :

15,14,17,14,13,14,15,12,13,15,17,17,12,11,14,12,19,15,17,13,18,15,16,17,17,15,13,14,20,15;

The number of different bits of feature vector between “浙’and others are :

14,17,12,9,16,13,14,15,10,14,10,18,11,12,13,15,14,12,14,9,9,10,15,12,14,15,10,11,16,15;

The number of different bits of feature vector between “青’and others are :

9,11,6,3,10,7,10,11,6,10,8,12,9,8,7,17,10,6,12,8,11,4,17,6,8,13,10,5,17,11;

The number of different bits of feature vector between “皖’and others are :

11,12,5,2,11,8,15,10,3,9,9,11,6,7,8,18,13,7,13,7,16,5,16,5,7,14,11,5,16,14;

The number of different bits of feature vector between “宁’and others are :

13,16,11,14,17,12,11,11,15,13,15,9,12,13,17,14,17,17,17,13,16,15,16,13,11,20,16,17,16,16;

The number of different bits of feature vector between “甘”and others are : 17,18,9,12,19,16,10,18,13,19,15,13,18,17,12,14,15,15,19,11,4,13,16,11,11,15,15,11,14,16.

The number of different bits of feature vector is also the Hamming distance between different characters. We could calculate noisy feature vectors if we know the minimum Hamming distanceη. Since some characters noisy samples are theoretically allowed are too much. we are here to set a theoretical premise, 1) Refuse to recognize character which is not close to any character. 2) fault-tolerant selected as 2. If the minimum Hamming distance between a character and others is over than 4, we should calculate 2-dimensional noisy samples which are theoretically allowed and then put them to the classifier which is designed according to the following method for training. If the minimum Hamming distance between a character and others is not more than 4, we should do secondary recognition.

3.4.2. Classifier Design

When training the standard samples (65 class characters), we must have more than one time samples. In this paper, we take 10 times samples. We design 68 classifiers. They are Chinese character classifier, digital classifier and alphabet classifier, other 65 classifiers for the other 65 class characters. The feature vector through the method of rough grid feature extraction is 35-dimensional, so we adapt a 3-layer BP neural network which contains a hidden layer, its input nodes is 35, output nodes is 1 and the number of neurons which hidden layer contained is different from each other. Therefore, in the actual design, the number of neurons which hidden layer contained can calculated by the empirical formula (15).

$$s = \sqrt{0.43mn + 0.12n^2 + 2.54m + 0.77n + 0.35} + 0.51 \qquad (15)$$

Through calculation and repeated experiment, we ultimately determined the number of neurons which contained by Chinese character network’s hidden layer is 20, the number of neurons which contained by alphabet network’s hidden layer is 16,and the number of neurons which contained by digital network’s hidden layer is 8 [7].

Firstly design 3 classifiers: Chinese character classifier, alphabet classifier and digital classifier. Take Chinese character classifier as an example. First training the BP neural network and demand that when input the 31 Chinese characters’ standard vector, the output is 1, otherwise the output is 0. The training is successful until the output error is less than 0.01. Design alphabet classifier and digital classifier as the same principle.

Secondly, design 65 character classifiers which means that design a classifier for each character. Take “鲁” as an example. First training the BP neural network and demand that when input the standard vector of “鲁”, the output is 1, otherwise the output is 0. The training is successful until the output error is less than 0.01. Input 2-dimensional noisy sample vectors which are theoretically allowed and other characters, then retrain the network and demand that when input 2-dimensional noisy sample vectors, the output is 1, otherwise the output is 0. The training is successful until the output error is less than 0.01. Figure 3 is the training result of “鲁” classifier without noise. Figure 4 is the training result of “鲁” classifier with noise.

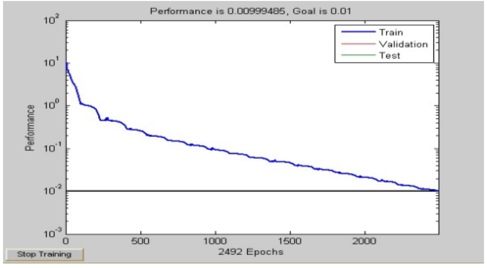

Figure 3. Training Result of “鲁” Classifier without Noise

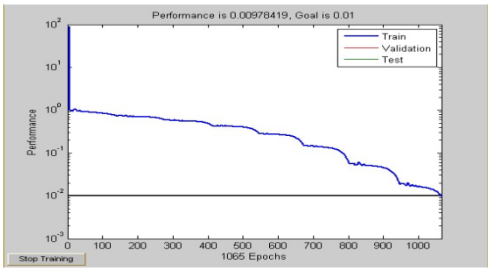

Figure 4. Training Result of “鲁” Classifier with Noise

3.4.3. Character Recognition

Save the successful neural network’s weight and bias matrix, and then get result through operation these matrix with unknown character. First input the unknown character to Chinese, alphabet and digital recognitions separately. If it is Chinese, input the character to 31 Chinese classifiers for recognizing, then we should get 31 floating-points which between 0 and 1. Take “鲁” as an example, the 31 floating-points are as follows:

(0.8673, 0.2231, 0.1765, 0.3943, 0.3754, 0.2487, 0.4612, 0.4752, 0.3211, 0.2513, 0.2435, 0.1823, 0.1286, 0.3341, 0.2714, 0.1123, 0.3522, 0.3478, 0.4013, 0.2231, 0.2463, 0.2847, 0.2645, 0.2336, 0.3156, 0.3746, 0.2568, 0.2931, 0.7156, 0.2460, 0.3025).

Compare these floating-points and we could find that 0.8673 is much bigger than others and it closes to 1. And 0.8673 is the output corresponding “鲁” classifier. So we can make a conclusion that the unknown character is “鲁”.

Character whose Hamming distance between it and standard “鲁” character is not more than 2 can be recognized as “鲁”. Character whose Hamming distance between it and others is more than 2 can be refused to recognize. If the consequent is not good enough, we should do secondary recognition.

3.4.4. Secondary Recognition

For these characters whose Hamming distance is not more than 4, the result is not so good by adopting the above method, so we do the secondary recognition. Take alphabet as an example, characters whose minimum Hamming distance is not more than 4 are D E F K Q R V W. We adopt 13-point feature extraction to extract the 8 characters’ feature. Design classifiers for the 8 characters. Then input the feature vectors to the BP neural network for training. Take “D” as an example, demand that when input “D” standard vector, the output is 1, otherwise the output is 0. The training is successful until the output error is less than 0.01. Next, save the successful neural network’s weight and bias matrix, and then get result through operation these matrix with unknown character. At the same time, also get some floating-points which between 0 and 1. Find the biggest point which means we could know the result. As an example, the "D", finally get a set of that number: (0.9858, 0.2312, 0.1432, 0.2897, 0.3813, 0.1783, 0.4332, 0.2526). 0.9858 is much bigger than others and close to 1, so we can conclude that the unknown character is “D”.

4. Conclusion

No matter normal training or noisy training, we could choose the same parameters. According to several experiments and the analysis of the network structure, we set the parameters as Table 2.

Table 2. The Initial Value of the Network Parameters

Network types	Learning rate	Momentum constant	Maximum cycles	Display interval	Output error
Chinese	0.08	0.9	5000	30	0.01
alphabet	0.08	0.9	5000	30	0.01
digital	0.08	0.9	5000	30	0.01

In the testing stage, we choose 100 testing samples for each classifier and the partial results are as Table 3.

Table 3. Part of the Test Results

Classifiers	Testing samples number	Correct number	Accuracy
“鲁”classifier	100	90	90%
“A” classifier	100	93	93%
“3” classifier	100	96	95%

Since use the color and shape information in the location stage, the location plate is more accurate which offers many advantages for the character segmentation and feature extraction. In the recognition stage, the accuracy has improved greatly because of using the improved BP neural network to recognize the fault-tolerant characters.

Although the BP algorithm has a solid theoretical basis and high versatility, it also has some weak points [8],

The improvement is as follows:

Optimize the initial weights. Select random number between 0 and 1 to be the initial weights in general.

Adjust learning rate adaptively. To ensure system's stability, select a smaller learning rate between 0.08 and 0.1.

Increase the momentum term. Momentum term has the effect of smooth and cushion, which could help improve the stability of the network's convergence process and also solve the problems of local minimum to some extent.

Acknowledgements

This work was financially supported by National Natural Science Foundation of China under Grant No. 61272077, 60973048.

References

[1] Frank Y, Shih, Souxian Cheng. Automatic seeded region growing for color image segmentation. *Image and Vision Computering*. 2005; 23: 877-886.

[2] Li Yucheng, Yang Guangming, Wang Mushu. Study and Design for System of License Plate Extraction and Recognition. *Computer Measurement and Control*. 2011; 19(1): 158-160.

[3] Hu Xiaofeng, ZhaoHui. Visual C++/MATLAB Image Processing and Recognition. Beijing: People Post Press. 2004: 94-101.

[4] Fan Weiqi, Mu Changjiang. A method of license plate character recognition based on Chinese character structure. *Instrument and Meter Journal*. 2003; 24(4): 472-474.

[5] Li Wenju, Liang Dequn. The algorithm of Quality degradation license plate character segmentation. *Computer Aided Design and Graphics Journal*. 2004; 6(5): 697-700.

[6] Wang Nian, Xiong Yuan, Zhao Haifeng, Ren Bin.Vehicle License Plate Recognition Automatically Based on Neural Network. *Anhui University Journal*, 2000; (3): 45-50. chapter, 10.

[7] Amir Sedighi, Mansur Vafadust. A new and robust method for character segmentation and recognition in license plate images. *Expert Systems with Applications.* 2011; 28(11): 13497–13504.

[8] Yin Zhaoqing, Yin Hao. Artificial Intelligence and Expert System. Beijing: China Water Resources and Hydropower Press. 2002.